

Available online at www.sciencedirect.com**ScienceDirect**

Procedia Computer Science 93 (2016) 774 – 781

Procedia
 Computer Science

 6th International Conference On Advances In Computing & Communications, ICACC 2016, 6-8
 September 2016, Cochin, India

Computations on Cipher Speech for Secure Biometrics

 Archana Dinesh^a, Edet Bijoy K^{b,*}
^aCalicut University, Department of Electronics and Communication Engineering, MES College of Engineering, Thrikkanapuram, Kerala, 679573, India

^bCalicut University, Department of Electronics and Communication Engineering, MES College of Engineering, Thrikkanapuram, Kerala, 679573, India

Abstract

Signal Processing in Encrypted Domain (SPED) is a blending of signal processing and cryptography. SPED is a magnificent tool for processing encrypted data. It provides privacy and security for highly confidential and sensitive data. In this paper we propose a system model for speech authentication, where authentication is done on cipher speech rather than raw speech in the data set. Here we have tested the proposed system with Advanced Encryption Standard (AES) ciphers and Rivest, Shamir and Adleman (RSA) ciphers with secure signal processing protocols like Secure Inner Product (SIP) and Secure Log Sum (SLS). The test results with AES ciphers and RSA ciphers with secure authentication protocols SIP and SIS are detailed in this paper. The experimental result shows that the SLS protocol works well for speech authentication irrespective of the encryption schemes and the feature selection. The performance of the system is evaluated using False Acceptance Rate (FAR) and False Rejection Rate (FRR) measures.

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of the Organizing Committee of ICACC 2016

Keywords: MFCC; PLP; Encoding; Hashing; AES; RSA; SIP and SLS protocols.

1. Introduction

Due to the rapid technological developments in the areas like social networking, online applications, cloud computing, and distributed processing, in general, have raised important concerns regarding the security and privacy but user related content. Use of biometric template for personal recognition¹ has been proved efficient and practical, but it raises several privacy issues. In particular, biometrics cannot be considered as secret data, biometric data are almost not revocable due to their permanent nature, whereas they are unique and can be used to identify someone among a large set of individuals.

Public environment is required to storing private data, but there are many security issues. To overcome this challenges we go for SPED, where the data is processed directly in the encrypted domain. Homomorphic encryption is an approach, in general, used to process data in encrypted domain. In 1978 Rivest, Adleman and Dertouzos³ intro-

* Edet Bijoy K. Tel.9037325763

E-mail address: pinkyachoo@gmail.com, edetbijoyk@ieee.org

duced privacy homomorphism. In this method homomorphism is used along with encryption, which has many security issues. In 1978 Rivest, Shamir and Adleman (RSA) proposed⁴ a public key cryptosystem, it is a multiplicative homomorphic encryption scheme. In RSA, multiplications can be performed with the encrypted data without decrypting the data. Goldwasser-Micali and Elgamal⁵ developed semantically secure additive homomorphic encryption scheme in 1985. Benaloh⁶ scheme in 1988 is a generalization of Goldwasser-Micali method. Naccache Stern⁷ scheme achieves smaller expansion of plaintext and thereby achieves superior efficiency. Pascal Paillier⁷ introduced additive homomorphic encryption scheme in 1999. It is a probabilistic asymmetric algorithm for public key cryptography.

In 2009 Craig Gentry⁸ proposed a fully homomorphic encryption scheme. It is the first encryption scheme to provide unlimited number of addition and multiplication on ciphertext. The fully homomorphic encryption scheme is developed from somewhat homomorphic encryption scheme. When homomorphic operation is applied on ciphertext, noise associated with it increases. When noise reaches a particular level the resulting ciphertext cannot be able to decrypt correctly. All existing fully homomorphic encryption schemes were developed from the Gentry's method. In 2010, Martin Van Dijk, Craig Gentry, Shai Haveli and Vinod Vaikundanathan⁹ developed a second fully homomorphic encryption scheme, which was an Integer based method deals with modular arithmetic operations. In 2012, Zvika Brakerski, Craig Gentry and Vinod Vaikundanathan⁹ developed a efficient somewhat and fully homomorphic cryptosystems, referred as second generation fully homomorphic encryption scheme. This method is based on learn with errors, having slower growth of noise during the homomorphic operations. Software library HELib⁹ for implementing homomorphic encryption was developed by IBM in 2013. Homomorphic Encryption is one of the most relevant types of encryption methods studied in the computational sciences today. All the techniques including fully, somewhat, and partially homomorphic encryption allows one to securely transmit, store, and process encrypted data without jeopardizing the confidentiality of data. The area of homomorphic cryptography remains an interesting area with a lot of scope for future research.

When considering biometric data, if the biometric templates were stolen, they may be used for illegal activities. In 2011, Sony's playstation¹ was hacked and the personal information of users were leaked and also discovered that Dropbox was storing user files in unencrypted format. These are motivations to prevent someone from learning the content of a remote database. SPED is the general signal processing tools that work directly on encrypted data, representing a valid solution for processing sensitive data by the third party. Processing signals in encrypted domain is an important challenge. Though many algorithms were developed based on homomorphic encryption schemes, the practical implementations of such schemes still remain a challenge. Hence new efficient and less complex homomorphic encryption based approaches can be developed for signal processing applications like privacy preserving speech processing and privacy preserving biometric authentication.

In this paper we deal with speech biometric data. The main objective of this paper is to develop a system model which will process the encrypted biometric signal in the encrypted domain while maintaining the biometric template protection, and store biometric template in encrypted domain. The main application of speech authentication is call steering, call centre authentication etc. Our contributions in this paper are summarized as follows:

- The existing works doesn't provides any real time implementation of homomorphic encryption for biometric system. Here we propose a new system model for secure biometric authentication.
- We propose a new secure speech authentication system, in which the features of the biometric data are encrypted using AES and RSA schemes and authentication is done using secure primitives like secure inner product and secure log sum. The proposed system is evaluated using False Acceptance Rate (FAR) and False Rejection Rate (FRR) measures.

The rest of the paper is organized as follows: In section 2, the proposed system model for privacy preserving speech processing is detailed, the secure signal processing approaches are given in section 3. The performance measures are explained in section 4. Implementation and simulation results are discussed in section 5 and section 6 respectively and the conclusion in Section 7.

2. System Model

Speech is one of the most simplest private forms of personal communication, speech sample contains information about the gender, accent and the emotional state of the speaker apart from the message content. This paper focusses on developing a privacy preserving speech verification framework. In this proposed method the system stores only encrypted speech samples of a speaker and it is used to authenticate the user. A generalized block diagram for privacy preserving speech processing is shown in Fig.1. Various steps in the privacy preserving speech processing are feature extraction, encoding, hashing, encryption, secure computations for authentication, compare with threshold and decision making.

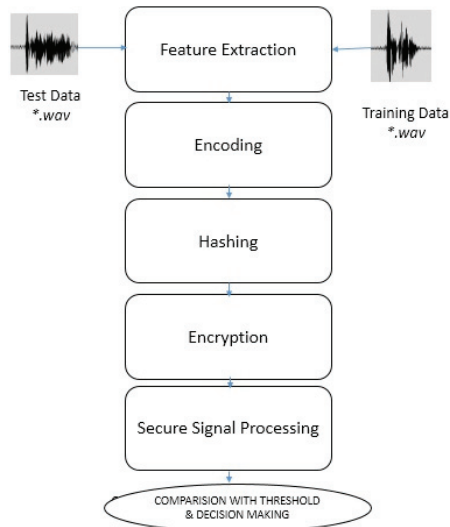


Fig. 1. Proposed system model

2.1. Feature extraction

To represent speech signals more compact and efficient, we go for best feature extraction models. In this case, we perform Mel Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Prediction (PLP).

2.1.1. Mel Frequency Cepstral Coefficients (MFCC)

Mel Frequency Cepstral Coefficient¹⁸ extract features according to human perception. Humans are less sensitive to frequency above 1KHz. Mel scale is linear up to 1 KHz and then varies logarithmically. Frequency to mel scale conversion is given in equation 1, where f is frequency in hertz.

$$m = 2595 * \log_{10}(1 + f/700) \quad (1)$$

The cepstral representation of speech provides characteristics of local spectrum.

2.1.2. Perceptual Linear Prediction (PLP)

PLP models the auditory systems spectrum using the autocorrelation linear prediction system. Main two treads involved in this method is obtaining auditory spectrum and approximating the auditory spectrum by an all pole model²¹. Auditory spectrum is derived from the speech waveform by critical-band filtering, equal loudness curve pre-emphasis, and intensity loudness root compression. Eighteen critical band filter outputs with their center frequencies equally spaced in bark domain, are defined in equation 2, where w is the angular frequency in rad/s.

$$\Omega(w) = 6 * \ln((w/1200 * \pi) + ((w/1200 * \pi)^2 + 1)^{1/2}) \quad (2)$$

2.2. Encoding

Analog transmission is much effected by noise. Pulse Code Modulation (PCM) is the technique used here to digitalize the samples. Levels of quantization is given in equation 3.

$$L = 2^n \quad (3)$$

Continuous samples are rounded off and truncated in to a finite range of discrete values. Discrete samples are represented in binary numbers having the range from 0 to n , where n is in powers of 2.

2.3. Hashing

A widely used cryptographic hash function to produce a message digest of 128 bit. The structure of this hash function is taken from Markle–Damgard construction. Three helper function involved .i.e. buffer, table and auxiliary function.

$$K_i = \text{abs}(\sin(i + 1)) * 232 * A \quad (4)$$

Table is used for improving computational speed. Total of 64 elements are inside it, each element is represented by equation 4.

2.4. Encryption

This paper mainly focused on AES and RSA cryptosystems. We use 128 bit standards, where the input to these encryption schemes is 128 bit. AES is a symmetric encryption scheme, here 128 bit plaintext converted into 128 bit ciphertext. Key sizes used by AES is 128, 192 and 256 bits¹⁷, having 10 rounds of operation, each round carrying four operations except for the last round all other rounds are identical. The four operations are substitute bytes, shift rows, mix columns and add roundkey. RSA is an asymmetric key cryptosystem, it uses both public and private keys. RSA's key size varies from 1024 to 4096 and it having a single round of operations. Each round involves four steps: key generation, key distribution, encryption and decryption. Public key (e, n) used for encryption, alike that a private key (d, n) is used for decryption.

2.5. Secure signal processing

To enable computations on highly confidential and sensitive data, i.e., to work directly on the encrypted data, we go for the secure signal processing approaches. Public key homomorphic encryption schemes, which enables some operations on the encrypted data. Complex mathematical operations can be done on the encrypted data without affecting the nature of the encryption. Homomorphic encryption consists of four functions; KeyGen, Encrypt, Evaluate and Decrypt. When decrypt the result of the evaluation algorithm, it gives the same result as if what operations done on the plaintext.

Various flavors of secure primitives for speech processing are secure inner product¹⁶, secure log sum, secure maximum value, secure maximum index¹⁴ etc. In this work we focused mainly on secure inner product (SIP) and secure log sum (SLS) protocols, it is shown in Table 1.

3. Secure Protocols

3.1. Secure inner product

Alice holds vector v and Bob vector w both lies in R^{d-1} , where $d \geq 3$. Alice is interested in computing $w^T \cdot v$ without revealing anything about her vector¹⁶. Bob is willing to participate in such a protocol as long as he does not reveal more than the dotproduct to Alice. Q is random $s \times s$ matrix, $s \geq 2$ where s is the security parameter. Randomly select

Table 1. Seure primitives utilized in this work

Primitives and Inputs	Output:Alice	Output:Bob	Relation
SIP(x,y)	a	b	$a+b=x^T y$
SLS(x,y)	a	b	$a+b=\ln \sum_{i=1}^d x_i + y_i$

$r, R_1, R_2, \text{ and } R_3$, where $1 \leq r \leq s$. Choose $s-1$ random $d \times 1$ vectors, x_i where $1 \leq i \leq s, i \neq r$. w' is vector in R^d for which $w'_i = w_i$, for all $1 \leq i \leq d-1$, and $w'_d = 1$. Set $x_r = w'$ and create an $s \times d$ matrix X , its i^{th} row is x_i^T ¹³.

$$b = \sum_{i=1}^s Q_{ir} \quad (5)$$

$$c = \sum_{i=1, i \neq r}^s (x_i^T * \sum_{j=1}^s Q_{ji}) \quad (6)$$

Bob chooses a random $1 \times d$ vector f ,

$$Q * X \quad (7)$$

$$c' = c + f^T * R_1 * R_2 \quad (8)$$

$$g = f * R_1 * R_3 \quad (9)$$

v' be in R^d such that $v'_i = v_i$ for all $1 \leq i \leq d-1$, and $v'_d = \alpha$, where α is a random number.

$$y = Q * X * v' \quad (10)$$

$$z = \sum_{i=1}^s y_i \quad (11)$$

$$a = z - c' * v' \quad (12)$$

Sends a to Bob and computes,

$$h = g^T * v' \quad (13)$$

$$\beta = (a + h * R_2 / R_3) / B \quad (14)$$

Send β to Alice and dotproduct is finally obtained as $\beta - \alpha$.

3.2. Secure log sum

Alice and Bob wish to obtain uninformative additive shares, a and b such that $a + b = \ln(\sum_{i=1}^d z_i)$ which is the log sum operation that gives the protocol its name. We note that and achieve the desired secret sharing using the following protocol²⁰,

1. Alice chooses a at random. Then Alice and Bob compute additive shares q, s such that $q + s = \text{SIP}(e^{x-a}, e^y)$ using the SIP protocol above. Bob combines these shares to obtain the inner product ϕ .
2. Bob computes $b = \ln \phi = -a + \ln(\sum_{i=1}^d e^{x_i+y_i}) = -a + \ln(\sum_{i=1}^d z_i)$ which gives the desired result.

In the first step above, Alice and Bob employ additive secret sharing in the exponent, which is equivalent to multiplicative secret sharing. The parameter a should be chosen large enough because multiplicative secret sharing is not as secure as standard additive secret sharing. We present this protocol to illuminate the fact that homomorphic functions can be manipulated to compute non obvious functions. The same functionality can be obtained in a secure manner using other cryptographic primitives. To refer this protocol, it will state that Alice and Bob obtain additive shares²⁰ a, b such that $a + b = \text{SLOG}(\ln z)$.

Table 2. Simulation results

Speech 1	Speech 2	AES				RSA			
		MFCC		PLP		MFCC		PLP	
		SIP	SLS	SIP	SLS	SIP	SLS	SIP	SLS
one1	one1	0.85	0.61	0.88	0.69	0.89	0.91	0.98	0.87
one1	one3	0.86	0.67	0.83	0.68	0.83	0.89	1	0.91
two1	two3	0.65	0.68	0.70	0.83	1	0.88	0.89	1
two1	two2	0.79	0.79	0.75	0.92	0.91	0.85	0.89	0.98
three1	three2	0.98	0.54	0.76	0.84	0.50	0.79	0.82	0.87
four1	four4	0.66	0.66	0.79	0.69	0.99	0.89	0.79	0.98
five1	five3	0.79	0.55	0.91	0.99	0.89	0.99	0.77	0.98
five1	five2	0.79	0.69	0.89	0.87	0.54	0.61	0.72	0.95
one2	one1	0.92	0.99	1	0.83	0.87	0.79	0.89	0.98
one3	one2	0.92	0.95	0.76	0.82	0.49	0.79	0.83	0.95
two2	two4	0.88	0.88	0.81	0.91	0.80	0.75	0.79	0.89
two1	two4	1	0.72	0.83	0.79	0.80	0.71	0.71	0.80
four4	four2	0.91	0.95	0.75	0.76	0.79	0.65	0.70	0.79
four2	four2	0.78	1	0.81	0.91	0.87	0.83	0.73	0.83
three1	three3	0.76	0.98	0.92	0.83	0.75	1	0.65	0.77
three2	three3	0.89	0.92	0.91	1	0.59	0.84	0.64	0.76
one1	two2	0.78	0.19	0.81	0.45	0.47	0.38	0.59	0.42
one1	five3	0.55	0.18	0.76	0.39	0.46	0.21	0.42	0.45
one1	six1	0.61	0.26	0.80	0.55	0.61	0.49	0.61	0.64
three1	seven2	0.60	0.12	0.57	0.52	0.82	0.59	0.39	0.55
three1	five2	0.68	0.24	0.58	0.53	0.59	0.69	0.35	0.65
two1	six2	0.55	0.14	0.57	0.19	0.39	0.23	0.63	0.61
two1	three1	0.56	0.29	0.54	0.18	0.41	0.19	0.64	0.64
five1	six1	0.69	0.23	0.49	0.50	0.54	0.39	0.65	0.63
two2	three2	0.65	0.25	0.79	0.48	0.52	0.42	0.65	0.52
two2	one1	0.62	0.26	0.75	0.50	0.53	0.48	0.59	0.53
one1	four1	0.66	0.18	0.58	0.55	0.53	0.19	0.42	0.43
two2	seven1	0.59	0.31	0.62	0.47	0.50	0.21	0.49	0.44
two3	six2	0.58	0.14	0.68	0.49	0.51	0.23	0.51	0.58
four2	one1	0.76	0.19	0.48	0.51	0.41	0.18	0.69	0.32
four2	five1	0.75	0.10	0.49	0.34	0.52	0.17	0.68	0.59
seven1	one3	0.83	0.18	0.44	0.30	0.59	0.13	0.58	0.29

4. Performance Measures

False Acceptance Rate (FAR) and False Rejection Rate (FRR) are measured to evaluate the system performance. FAR and FRR are defined in equation 15 and 16 respectively.

$$FAR = \frac{\text{Impostor scores exceeding threshold}}{\text{All impostor scores}} \quad (15)$$

$$FRR = \frac{\text{Genuine scores failing below threshold}}{\text{All genuine scores}} \quad (16)$$

Table 3. Performance measures

Performance measures	AES				RSA			
	MFCC		PLP		MFCC		PLP	
	SIP	SLS	SIP	SLS	SIP	SLS	SIP	SLS
FAR	0.31	0	0.19	0	0.31	0.06	0.38	0
FRR	0.13	0	0.31	0	0.06	0.06	0.13	0

5. Implementation

The implementation is done in MATLAB R2014a. The MFCC and PLP features are extracted from speech signal. Then encoded using PCM, then the arbitrary length binary data is converted into 128 bit data by using MD5 hash function. Then the 128 bit feature is encrypted using AES and RSA cryptosystems. Then correlation between cipher speeches are measured using the secure inner product and secure log sum protocols.

6. Simulation Results and Discussions

Here we implement a secure speech authentication system. The extracted features from different speech samples are encrypted. This includes speech samples of different speakers spoken at different tones. In Table 2. the speech sample one1 and one2 corresponds to same speech (Malayalam digit one) spoken at different tones. Table 2. shows the experimental results of different encrypted speech samples whose MFCC and PLP features are encrypted using AES and RSA separately and authentication done using SIP and SLS protocols.

The results shown here are normalized SIP and SLS scores. From the obtained results, we can clearly observe that the scores obtained by SLS protocols is the best irrespective of the encryption scheme and feature selection. The scores corresponding to positive acceptance stand above the threshold and the scores corresponding to rejection fall far below the threshold. Performance evaluation of the system using FAR and FRR scores is depicted in Table 3. From the results it is observed that SLS protocol performs well for both AES and RSA and for any feature chosen as well.

7. Conclusion

In this paper, we have proposed a privacy preserving speech authentication system. We used MFCC and PLP to extract features from the speech. Secure inner product and secure log sum signal processing primitives were used to implement the proposed system, where these primitives were applied to encrypted features of speech data, encryption where done using AES and RSA schemes. Experimental results show that secure log sum outperforms secure inner product for speech authentication system irrespective of the feature chosen and the encryption schemes.

References

1. Ogburn M, Turner C, Dahal P. Homomorphic encryption. *J. Elect. Procedia Comput. Science.*.. Rolla, USA, vol.20, May.2013, p.502-509.
2. Van Dijk M, Gentry C, Halevi S, Vaikuntanathan V. Fully homomorphic encryption over the integers. in *EUROCRYPT*. 2011; p.24-43.
3. Rivest RL, Adleman L, Miachel Dertouzos L. On data banks and privacy homomorphisms. *J. of Foundations of secure computation*. vol. 4, no. 11, p.169-180. Oct.1978.
4. Parmar VP, Padhar BS, Patel NS, Jhaveri HR. Survey of various homomorphic encryption algorithms and schemes. *Int. J. of Comput. Applications*. vol.91, no.8, Apr. 2014.
5. Goldwasser S, Micali S. Probabilistic encryption and how to play mental poker keeping secret all partial information. in *Proc. of the 4th annual symposium on Theory of comput. (ACM)*. may.1982; p. 365-377.
6. Fousse Laurent, Lafourcade, Pascal P, Alnuaimi M. *Progress in Cryptology (AFRICACRYPT)*. Springer. 2011; p.348-362.
7. Pascal P. Trapdoor discrete logarithms on elliptic curves over rings. *Advances in Cryptology (ASIACRYPT)*. Springer. 2000; p.573-584.
8. Gentry C. A fully homomorphic encryption scheme." Ph.D. dissertation. Stanford Univ. New York: USA; 2009.
9. Ducas L, Micciancio D. FHEW: Bootstrapping homomorphic encryption in less than a second. *Advances in Cryptology (EUROCRYPT)*. Springer. 2015; p.617-640.

10. Brakerski Z, Vaikuntanathan V. Efficient fully homomorphic encryption from (standard) LWE. *SIAM J. on Computing*. vol.43. no.2. p.831-871. Apr. 2014.
11. Moore C, Maire Neill, Sullivan E, Doroz Y, Sunar B. Practical homomorphic encryption: A survey. *IEEE Int. Symposium (ISCAS)* . p.2792-2795. 2014.
12. Sadkan Sattar B, Abdulaheem HF. An analytical study for security evaluation of cryptosystems used in cloud networking. *Proc. IEEE Int. Conf. (ICECCPCE)*. Mosul Univ: Iraq; Dec. 2013; p. 157 - 162.
13. Bianchi T, Alessandro Piva, Mauro Barni. On the implementation of the discrete Fourier transform in the encrypted domain. *IEEE Trans. Information Forensics and Security*. vol.4. no.1. p.86-97. Mar. 2009.
14. Manas Pathak A, Raj B, Rane S, Smaragdis P. Privacy Preserving Speech Processing. *IEEE Signal Process. Magazine* . vol.30. p.62-74. Mar. 2013.
15. Reginald L, Erkin Z, Barni M. Encrypted signal processing for privacy protection Conveying the utility of homomorphic encryption and multiparty computation. *J. IEEE Signal Process. Magazine*. vol.30, no. 1. p. 82-105. Jan. 2013.
16. Ioannidis I, Grama A, Atallah M. A secure protocol for computing dot products in clustered and distributed environments. *Proc. IEEE Int. Conf. on Parallel Processing* . Dept. of Comput. Sciences. Purdue Univ. West Lafayette: IN; 2002.
17. Joan Daemen, Rijmen V. Advanced Encryption Standard. "Federal Information Processing Standards Publication. 1st ed. U.S.; Nov. 2001.
18. Ching Wang M, Fa Wang J, Weng Y. Chip design of MFCC extraction for speech recognition. *J. of VLSI INTEGRATION*. vol.32. no.1. p.111–131. 2002.
19. Nan Hu. Secure image processing. Ph.D. dissertation. Dept. Computer Science. Eng. Kentucky Univ. Lexington: US; 2007.
20. Shashanka MA. privacy preserving framework for gaussian mixture models. *Proc. IEEE Int. Conf. (ICDMW)* , p.499-506. 2010.
21. Hermansky H. Perceptual linear predictive (PLP) analysis of speech. *the J. of the Acoustical Society of America* . Vol.87. no.4. pp.1738-1752. 1990.